THE EFFECTS OF SCREENCASTING ON STUDENTS' ACHIEVEMENT AND ATTITUDE TOWARDS C++ COMPUTER PROGRAMMING

CHEAH CHIN SOON

UNIVERSITI SAINS MALAYSIA

2018

THE EFFECTS OF SCREENCASTING ON STUDENTS' ACHIEVEMENT AND ATTITUDE TOWARDS C++ COMPUTER PROGRAMMING

by

CHEAH CHIN SOON

Thesis submitted in fulfillment of the requirements
for the degree of
Doctor of Philosophy

June 2018

ACKNOWLEDGEMENT

First of all, I would like to express my appreciation and gratitude to my supervisor, Dr. Leong Lai Mei for her precious advice, patience, support, and time in guiding me to complete this thesis. I could not have come this far and succeed in producing this thesis without her amazing support and motivation. I would also like to thank my co-supervisor Professor Dr. Nordin Abd. Razak for his professional expertise in statistic advice and time in guiding me. His strong knowledge in statistic and SPSS has been extremely helpful in guiding me to conduct good analysis on the data that I have collected.

Secondly, I would like to express my gratitude to the Dean of the School of Educational Studies, Universiti Sains Malaysia, Professor Dr. Hairul Nizam Ismail, Deputy Dean Professor Dr. Muhammad Kamarul Kabilan Abdullah and the school for providing me a conducive learning environment with all the necessary facilities. I would also like to express my appreciation to the administrative staff which has provided me continuous support and assistance. At the same time, I would also like to thank Associate Professor Dr. Azidah Abu Ziden and Dr. Amelia Abdullah for the important comments and advice that they have given me. I would like to extend my acknowledgement to Professor Dr. Fong Soon Fook for his guidance during the early stage of my study and Associate Professor Dr. Chew Cheng Meng for the relevant information pertaining to my research.

My appreciation and gratitude also goes to the Dean of the School of Computer Sciences, Universiti Sains Malaysia, Professor Dr. Ahamad Tajudin Khader and Deputy Dean Associate Professor Dr. Zurinahni Zainol for allowing me to conduct the research in the School of Computer Sciences. I further extend my thanks to the

lecturers that were involved with and assisted me throughout the entire research. I would also like to thank the technical staff that have provided me support and guidance pertaining to the facilities in the computer lab.

Last but not least, I am grateful to my family and friends which have provided me moral and emotional support during my course of study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

KESAN *SCREENCASTING* DALAM PENCAPAIAN DAN SIKAP PELAJAR TERHADAP PENGATURCARAAN KOMPUTER C++

ABSTRAK

Tujuan kajian ini adalah untuk mereka bentuk, membina dan menilai keberkesanan *screencast* dalam pembelajaran pengaturcaraan komputer C++ dalam kalangan pelajar. Kajian ini juga mengkaji kesan prinsip *modality* dan *redundancy* dalam pencapaian dan sikap terhadap logik yang berbeza dalam kalangan pelajar. Kajian ini dijalankan berdasarkan reka bentuk eksperimen sebenar *pretest* dan *posttest* antara faktor subjek 2 x 2. Pembolehubah bebas adalah terdiri daripada dua mod pembelajaran iaitu mod *screencast* dan narrasi (SN), dan mod *screencast*, teks dan narasi (STN), manakala pembolehubah bersandar adalah terdiri daripada ujian pencapaian (ujian pengaturcaraan C++) dan skala sikap komputer (CAS). Sementara itu, pemboleh ubah moderator adalah tahap logik yang berbeza (Rendah / Tinggi) dalam kalangan pelajar. Sampel terdiri daripada 65 mahasiswa dari tahun pertama Program Sains Komputer di sebuah institut pendidikan tinggi dan dipilih berdasarkan kriteria yang mereka tidak pernah menghadiri kursus pengaturcaraan komputer rasmi sebelum ini. Selepas itu, mereka diasingkan secara rambang ke salah satu daripada dua mod pembelajaran (SN atau STN). Data yang diperolehi daripada kajian ini dianalisis dengan melakukan analisis kovarians (ANCOVA) untuk menentukan perbezaan yang signifikan di antara dua mod pembelajaran. Sementara itu, korelasi *Pearson* juga dijalankan untuk menentukan hubungan antara sikap dan pencapaian pelajar dalam dua mod pembelajaran itu. Teori yang digunakan dalam kajian ini ialah Teori Kognitif *Mayer Multimedia Learning* (CTML) dan model yang membimbing pembinaan kedua-dua mod pembelajaran ialah Model Reka Bentuk Pengajaran *ASSURE*.

Keputusan menunjukkan terdapat perbezaan yang signifikan dalam pencapaian pelajar dengan menggunakan dua mod pembelajaran tersebut. Pelajar yang menggunakan mod SN memperolehi pencapaian yang jauh lebih baik berbanding dengan pelajar yang menggunakan mod STN. Dari segi memupuk sikap positif terhadap pembelajaran pengaturcaraan C++ dalam kalangan pelajar, hasil perbezaan yang signifikan didapati. Mod STN didapati memupuk lebih sikap positif berbanding dengan mod SN. Walau bagaimanapun, kesan interaksi yang tidak signifikan didapati antara tahap logik yang berbeza (Rendah / Tinggi) dan dua mod pembelajaran SN dan STN. Untuk menentukan hubungan antara sikap pelajar dan dua mod pembelajaran, analisis korelasi *Pearson* dilakukan. Keputusan daripada korelasi *Pearson* menunjukkan bahawa terdapat hubungan positif yang lemah antara sikap pelajar terhadap pembelajaran pengaturcaraan komputer C++ dan pencapaian dalam pembelajaran pengaturcaraan komputer C++ dengan menggunakan dua mod pembelajaran tersebut.

THE EFFECTS OF SCREENCASTING ON STUDENTS' ACHIEVEMENT AND ATTITUDE TOWARDS C++ COMPUTER PROGRAMMING

ABSTRACT

The purpose of this study was to design, develop and evaluate the effectiveness of screencast in the learning of C++ Computer Programming among undergraduates. The study also investigated the effects of modality and redundancy principles on the achievement and attitude of the different levels of logic among the undergraduates. The study was conducted based on true experimental pretest and posttest 2 x 2 between subject factorial design. The independent variable consists of the two learning modes namely, screencast and narration mode (SN), and screencast, text and narration mode (STN), while the dependent variables are the achievement test (C++ Computer Programming test) and computer attitude scale (CAS) score. Meanwhile, the moderator variable was the different levels of logic (Low/High) among the students. The sample consists of 65 undergraduates from the first year Computer Science program at a tertiary education institute and were chosen based on the criteria that they had never attended any official computer programming course before. After that, they were randomly assigned to one of the two learning modes (SN or STN). The data obtained from this study were analysed by conducting analyses of covariance (ANCOVA) to determine the significant difference among the two learning modes. Meanwhile, Pearson correlation was also conducted to ascertain the relationship between the attitude and the students' achievement in the two learning modes. The theory used in this study is the Mayer's Cognitive Theory of Multimedia Learning (CTML) and the model that guided the development of the two learning modes is the ASSURE Instructional Design Model. The results showed that there was a significant

difference in the students' achievement in using the two learning modes. The students using the SN mode performed significantly better in their achievement compare to the students using the STN mode. In terms of instilling positive attitude towards the learning of C++ computer programming among the students, a significant difference result was obtained. The STN mode was found to instil more positive attitude compare to the SN mode. However, a non-significant interaction effects were found between the different levels of logic (Low/High) and the two learning modes SN and STN. To ascertain the relationship between the students' attitude and the two learning modes, the Pearson correlation analysis was performed. Results from the Pearson correlation have shown that there was a weak positive correlation between the students' attitude towards learning of C++ computer programming and the achievement in the learning of C++ computer programming using the two learning modes.

CHAPTER 1

INTRODUCTION

1.1 Introduction

The integration of multimedia into teaching and learning has played an important role in the current learning environment especially to the Net generation (Barnes, Marateo, & Ferris, 2007; Beyers, 2009; Hargittai, 2010; Neo & Neo, 2001). It has altered the traditional way of instructions given by teachers to students in passive learning that has been used for decades. The shift of pedagogy from teacher-centred to student-centred is the direction in the current emerging technology environment.

In 1991, the Malaysian government has shown that the involvement of multimedia in education is definitely inevitable when they unveiled Vision 2020 as the year in which Malaysia would achieve the status of industrialized and developed country (Mahathir, 1991). The government has highlighted nine challenges, and one of the challenges is to establish a scientific and progressive society, a society that is innovative and forward-looking. Furthermore, Malaysians will no longer just be a consumer of technology but also a contributor to the scientific and technological civilisation of the future. In order to achieve these goals, the society needs to be active learners and possess problem-solving skills, independent thinking and co-operatively interacting with other individuals.

Moreover, in 2003 the government has supported the use of Information Communication Technology (ICT) in the teaching and learning environment in the classroom. These include tools such as laptops, LCD projectors, speakers, software such as PowerPoint, and interactive programs like Flash to support teaching and learning processes. With the availability of ICT, this will enhance the interactivity of

the teaching and learning experience by using advanced multimedia and graphics to capture the student's attention (Lau & Sim, 2008).

In order to become a contributor to the scientific and technological civilisation, one of the important knowledge and skills required is the Information Communication Technology (ICT) skills, which includes computer programming (Islam, 2010). In Malaysia, computer programming is a compulsory subject in the field of Information Technology. It has been one of the toughest subjects for educators to teach and to be understood by the students (Ismail, Ngah, & Umar, 2010b; Tan, Ting, & Ling, 2009). Therefore, multimedia is opt as the tool to teach computer programming to the Net generation.

Thus, multimedia educations plays an important role in the 21st century in developing a knowledge-based society and achieving Malaysia's goals. The multimedia technology involves the usage of sounds, graphics, images, videos and animations in a computer-based presentation. Besides that, it is also able to include links and tools that allow the user to interact and navigate around the program (Barnes et al., 2007; Hofstetter, 2001; Sankey, Birch, & Gardiner, 2011).

The current generation lives in a visual animation world that is challenge with enhanced visual images and graphics that can be imagine in any possible way (Barnes et al., 2007; Lin & Atkinson, 2011; Sankey et al., 2011). By using multimedia technology to create a learning environment, various methodologies such as quizzes, tests, tutorials, web-based learning, drills, and exercises can be implemented in an interactive environment. This in turn will create a new dimension of learning environment that is not constrained by space, time and limitations of traditional

teaching methods (Barnes et al., 2007; Berk, 2009; Dong & Li, 2011; Malik & Agarwal, 2012).

This is further supported by the research done by Morain and Swarts (2012), whereby the Net Generation fills their own learning gaps by browsing the Internet for information from Google, YouTube and Wikipedia. They are no longer restricted to traditional learning materials such as textbooks or PowerPoint slides provided by their educators. Massive information can be found online at anytime and anywhere without being constrained within the classroom environment.

Hence, one of the multimedia elements which has gained popularity in higher education is screencasting (Mohorovičić & Tijan, 2011). Screencast is defined as a digital recording of computer screen activity and usually accompanied by audio narrations (Udell, 2004). This method of teaching has been utilised, shared and published via YouTube and Vimeo (Mohorovičić & Tijan, 2011). It has gained a lot of popularity and showed great potential in the education world due to its similarity to YouTube which the Net Generation is in favour of.

Moreover, a plethora of research has been done to investigate the effectiveness of screencasting in various fields of education. Examples of early success in the adoption of screencasting in the education field are in mathematics and engineering (Ashdown, Doria, & Wozny, 2011; Carr & Ly, 2009; Ghilay & Ghilay, 2015; Jordan, Loch, Lowe, Mestel, & Wilkins, 2012; Peterson, 2007; Soto & Schwappach, 2016; Trenholm, Alcock, & Robinson, 2012; Wakeman, 2013), Thus, screencasting is opted by the researcher as the teaching tool for creating computer instruction materials in this research.

To further enhance and to investigate the effectiveness of screencasting in the teaching of computer programming, the researcher has decided to dwell deeper into other principles as well. The screencast is developed not only to teach computer programming, but is also designed and developed to investigate the effects of modality and redundancy principles as well. Past research has shown that modality and redundancy effects occur in most of the scenarios (Dousay, 2016; Ritzhaupt, Pastore, & Davis, 2015; Schüler, Scheiter, & Gerjets, 2013; Smith & Ayres, 2016).

There are exceptional cases where the redundancy principles were opposed. Critiques have highlighted that if learners' do not learn in their native language, learning of foreign languages, complicated technical terms, and diagram were shown, the on-screen text is an aid to the beginners to better understand and learn. This has shown the opposing results of the redundancy effects (Kalyuga, Chandler, & Sweller, 2004; Mayer & Clark, 2011; Sorden, 2005).

Therefore, the aim of this study is to design, develop and evaluate the effectiveness of multimedia learning materials using the screencasting method in the learning of introductory to C++ computer programming lesson. The focus of this study is to determine the effectiveness of the multimedia learning environment both as a tool and instructional medium to increase the students' performance and achievement in learning C++ computer programming. Besides that, this research also investigates the effects of modality and redundancy principles as well.

1.2 Background of Study

We are heading towards the advancement of the technology Era in the 21st century. No matter where we go or what we do, we are surrounded by electronic gadgets that serve to assist and ease our daily lives. Thus, one of the core competency skills in this century is the Information Communication Technology (ICT). ICT is not just a tool for supporting acquisition of knowledge, but also a tool for the assessment of skills in various fields (Voogt & Roblin, 2010). Besides that, the growth of population having high ICT skills will contribute towards the development of a country's economy and human resources.

ICT includes skills and knowledge in software that is also known as computer program. Computer program is a set of machine-readable instructions that directs a computer to perform a specific task to achieve its objective. Hardware will only work if the appropriate software is installed in a particular electronic device. The synchronisation of both hardware and software will only achieve its goals and produces the desired result if a well-designed and coded software program has been installed and used. Therefore, education in programming skills is the foundation in developing skilleds programmers to achieve the overall goals and objectives (Gomes & Mendes, 2007).

The current generation generally has a shorter attention span when it comes to repetitive and boring static media (Velički & Velički, 2015). In traditional teaching methods, the use of static materials such as lecture notes, textbooks and slide presentations are ineffective in teaching computer programming (Bennedsen & Caspersen, 2005). The way for delivering knowledge is constrained by time, space and media available. Chalk, blackboard, and figure of speech are the primary ways to

conduct classroom education (Jenkins, Goel, & Morrell, 2008). Teachers play an important role as the expert on a particular subject that provides the main source of knowledge and reference to the students. The medium that is used to support teaching is mainly textual content which are the physically printed books carried by the students (Bennedsen & Caspersen, 2005).

Currently, modern education is moving towards developing problem-solving skills, creative thinking, decision-making, critical thinking, analyses, and evaluations (Bates, 2000). These are skills that are needed in order to master in computer programming skills successfully. Thus, a change in the standard education system that can support customisations to meet all learners' needs is required (Sancho, 2009).

Besides that, constructivist learning pedagogy has been identified as one of the effective methods in learning computer programming (Chetty & Barlow-Jones, 2014). Constructivist learning process has been described as learners actively engaging in investigating, exploring and problem-solving by creating personal interpretation of the world in constructing new meanings based on their current knowledge, past experience and social environment (Cakir, 2008; Cobb & Steffe, 2010; Neo & Neo, 2009; Wang, 2008).

In order to make the usage of technology effective in education, a move towards student-centred constructivist approach is needed (Tom, 2015). A teacher's role has changed from a knowledge expert to a facilitator, motivator, supporter, and guide to assist learners to actively engage and interact in a multimedia environment as the new way to acquire knowledge.

In terms of cognitive aspect, logic reasoning is required in understanding computer programming (Doets, 2012; Wegerif, 2002). Past research has shown that

there is a positive correlation between computer programming and general reasoning (Eckerdal, Thuné, & Berglund, 2005; Gomes & Mendes, 2007; White & Sivitanides, 2003). The understanding of the logic in the "if-then-else" statements and subprocedures requires the ability to do Propositional Logic, which coexists within Piaget's Cognitive Development formal operations (Brainerd, 1978; Enyeart, 1983; Irons, 1982).

According to the research done by Ismail et al. (2010b), a lack of logical and reasoning skills is one of the main contributors to the failure in understanding computer programming. He believes that, students should be introduced to courses pertaining to understanding logic before attempting to enrol in computer programming subjects. Failure to grasp the logical thinking skills and reasoning strategies leads to the creation of logic errors during the development of computer programs. Furthermore, logic errors are crucial and difficult to identify, as it will produce an output but with incorrect values.

Besides that, students' attitude and perception on computer programming also play an important role in terms of intrinsic motivation. Attitude determines one's behaviour and cognition that signifies a person's actions, which will determine the success or failure of an individual in achieving a certain task. Past research carried out by Tüysüz (2010) shows that there is a positive correlation between attitude and achievement. Furthermore, positive correlation between the students' attitude and their achievements in computer programming were also found in the research conducted by Baser (2013).

Computer programming is a complex subject that requires continuous effort, special approach and multi-layer skills. The process of obtaining these skills is a

tedious trial-and-error process and requires persistency (Jiau, Chen, & Ssu, 2009). A series of abilities and persistency is needed to become a good programmer and knowing the syntax of programming is just the beginning of the challenging aspect in creating a good program. Students need to understand beyond the syntax and flow of computer programming in order to resolve complex real world problems (Holvikivi, 2010).

In terms of teaching syllabus, instructors are more focused on teaching the syntactic details of programming rather than promoting problem-solving methods. In addition to that, the preferred programming language is chosen largely based on its popularity in the industry rather than pedagogic purposes (Gomes & Mendes, 2007).

According to Tan et al. (2009), in Malaysia, programming is a compulsory subject in the field of Information Technology, and it has been a tough subject for instructors to teach and for students to learn. The main difficulties in the learning of programming are designing a program to solve certain tasks and dividing functionality into procedures. Besides that, students encounter difficulties when learning the syntax of the language and eliminating bugs in the program, which is a time-consuming, and tedious process (Tan et al., 2009).

The research done by Tan, Ting, and Ling (2009) was further dwelled into investigating the reasons behind the difficulties in the learning of computer programming. The respondents agreed that they have problems designing a program to solve a certain task. This is due to the reason that they find it hard to divide the functions into procedures. This evidently proves that students face a lot of problem during the initial stage of understanding functions, procedures and programs control. Such problems existed a long time ago and the findings correspond with past research

by Spohrer and Soloway (1986). Novice programmers tend to merge different processes into a same piece of code when it should be implemented separately. Similar confusions happens when novices have knowledge of the syntax and semantics, but fail to understand how to combine these features into a successful program (Winslow, 1996).

Therefore, success in the learning of computer programming is determined by many factors. The types of learning materials, cognitive aspects in terms of logic reasoning and personal attitudes are the few main variables that contribute to the success of understanding computer programming. In terms of teaching materials, most of the problems faced by the students in the learning of computer programming is due to the traditional teaching method that uses static material (Bennedsen & Caspersen, 2005; Butler & Morgan, 2007; Robins, Rountree, & Rountree, 2003).

Hence, one of the teaching methods that fulfils most of the requirements in the learning of computer programming is by using screencast (Bennedsen & Caspersen, 2005). Screencast is defined as a digital recording of computer screen output or video screen capture and often contain audio narration as well (Udell, 2004). Screencast consists of dynamic elements that support spatial and visual capabilities that accommodate the nature of learning computer programming in terms of logic reasoning. It enables the students to have a mental representation of a problem and assist the students in understanding the process control and data flow (Bennedsen & Caspersen, 2005). Besides that, students are also allowed to control the screencast, such as forward, rewind, pause, or stop according to students' preferences. In terms of audio narration, students are given the option to adjust the volume according to their background environment. This accommodates the different learning pace of each student.

Moreover, the Net Generation prefers to learn by browsing the Internet for information, such as via Google, YouTube, Vimeo and Wikipedia at anytime and anywhere without being constrained in a classroom environment (Morain & Swarts, 2012). Hence, screencast has gained popularity due to its similarity to YouTube and Vimeo. Therefore, the researcher has decided to use this teaching method for the research. Besides, screencast also rectifies the tedious learning process of using static materials and provides a conducive learning environment. It motivates the student, produces positive attitude towards learning and enhances the student's learning experience (Lloyd & Robertson, 2012).

1.3 Statement of Problem

In this era of advanced technology, most of our daily tasks involve the use of computer devices. These devices will only perform its specific tasks and achieve user objectives when appropriate computer programs are installed in these devices. Thus, computer programming is the fundamental skills and knowledge required to fulfil the demands of high quality and effective programs (Martínez-Frías, 2003). By successfully implementing an effective ICT infrastructure, this will contribute to the success of our country in achieving the country's vision of attaining the status of industrialised and developed nation (Islam, 2010). One of the challenges highlighted in Vision 2020 is to become a scientific and progressive society that is innovative and forward-looking. By 2020, Malaysians will not only be a consumer of technology but also a contributor to the scientific and technological civilisation of the future (Mahathir, 1991).

Globally, learning computer programming has been a universal problem in the computer science curriculum. Computer programming subjects generally have high failure rates and withdrawals even at the initial stage of introductory courses (Butler & Morgan, 2007; Robins et al., 2003). There are many tools available that can assist the teaching and learning of programming, such as books and presentation tools used by educators, but the problems still remain unresolved (Tan et al., 2009). On the aspect of students' knowledge, many of them lack of problem-solving abilities (Gomes & Mendes, 2007) coupled with negative attitude towards the subject (Baser, 2013).

According to a research conducted by Gomes and Mendes (2007), it has shown that problems occur at the initial stage of learning programming in understanding abstract programming concepts such as control structures and creating algorithms to solve concrete problems. Consolidation of knowledge and abilities in problem-solving and logic reasoning is also needed to enhance the understanding of this subject (Hadjerrouit, 2008).

Traditional teaching methods using static materials such as printed books and PowerPoint slides do not seem to be an effective method in the teaching of programming (Bennedsen & Caspersen, 2005) as it does not provide live interactions. It would be ideal that an instructor is available to give immediate feedback and detailed explanation as and when required by the students. However, this is impossible due to time constraints as well as the number of students available in a single class. Moreover, static materials used in the teaching of computer programming might have impaired the learning process. These materials are unable to explain effectively the dynamic concepts of programming. For some students, they fail to understand the dynamic nature of programs whenever static materials are used to explain to them (Gomes & Mendes, 2007).

According to Jenkins (2002), existing teaching methods using static materials do not cater to the learning style of every students. He explained that some students might prefer learning alone whereas some might prefer a dynamic learning environment such as group discussion and interaction among peers. Most of all, a dynamic teaching material is needed to create a conducive learning environment (Sarpong, Arthur, & Amoako, 2013). Thus, it is important that the instructor ensures that the teaching methods provided are able to accommodate the different groups of students.

In addition, psychologically students also have "negative connotation" relating to programming subjects (Law, Lee, & Yu, 2010). This is due to the passing of comments, opinions, and suggestions from their senior peers who have previously taken these subjects (Jenkins, 2002). With the negative image and impressions, students tend to believe that programming is difficult (Rogerson & Scott, 2010). By having negative perceptions, it influences the students' attitude on computer programming.

An individual's attitude signifies the cognitive action that determines the success or failure in achieving certain tasks. Past research shows that there is a positive correlation between attitude and achievement in subjects such as biology and chemistry (Baser, 2013; Prokop, Tuncer, & Chudá, 2007; Tüysüz, 2010).

However, some research obtained opposing correlation results. A study conducted by Das, Halder, and Mishra (2014) on the correlation between attitude and achievement in secondary level students show a weak negative correlation which was not statistically significant. In another research done by Mubeen, Saeed, and Arif (2013) on attitude and mathematics achievement, a non-significant value was attained.

Similar results were also found in the study between attitude and biology achievement (Nasr & Soltani, 2011). A study conducted by Selçuk (2010) on the attitude and Physics achievement show a non-correlation result as well. Thus, it has sparked an interest to the researcher of this study to ascertain the correlation between the attitude toward learning of computer programming and achievement in computer programming.

Researches carried out in Malaysia have identified four main problems in the computer programming education in Malaysia (Ismail et al., 2010b). First, is the lack of skills in analysing problems and understanding programming concepts. This is due to the reason that students lack the prerequisite skills such as logic programming. Second, is the ineffective use of presentation techniques for problem-solving. The usage of conventional techniques such as pseudo code and flow chart are unsuitable for teaching computer programming. At present, most of the programming languages require more visualisations to enable the students to have a mental representation of the problem. The third problem arises from the ineffective use of teaching strategies for problem-solving and coding (Ismail et al., 2010b). Conventional teaching strategies that are teacher-centred are no longer effective in programming. Educators who teach computer programming agree that different paradigms in terms of cognitive strategy should be used in teaching programming. The use of teaching materials that supports spatial and visualisation abilities will assist the students in understanding the process of control and data flow. The final contributor of failure in the learning of programming is that students generally do not understand or master the programming syntax and constructs well. This can be further elaborated in terms of understanding the concepts of programming, lack of knowledge in syntax, and inability to construct effective programming codes (Ismail et al., 2010b).

Having identified all the contributors of failure in the learning of computer programming that persist until present time, this has created interest and doubts that motivates the researcher to conduct this study to explore and to discover better solutions in teaching computer programming. The identification of all the contributors above is necessary as it contributes to the effectiveness of applying the correct computer programming logic in developing a good computer program. Moreover, identifying the other contributing factors such as programming syntax and programming constructs further assist in developing the content of the two screencasts (SN and STN).

Therefore, the researcher has decided to implements two types of multimedia learning modes, which are the screencast and narration (SN) mode, and screencast, text and narration (STN) mode. The researcher has opted for screencast as the ideal learning modes because it has the dynamic element that is able to explain the dynamic nature of computer programming concepts. The study determines the effectiveness of these two learning modes on the learners' achievements. Furthermore, the study also investigates the learners' logic understanding and attitude towards the learning of computer programming. Besides that, the purpose of having the two different modes is to examine the effect of modality and redundancy principle as well. Pearson correlation coefficient is conducted as well to determine the degree of the relationship between the two variables.

1.4 Research Objectives

The main aim of this study is to examine the effectiveness of screencast in the teaching of C++ computer programming on students' achievement and attitude.

Besides that, it is also to examine the effects of the modality and redundancy principles on the learning of C++ computer programming using computer-assisted instructions. The objectives of this study are therefore as follows:

1. To design, develop, and implement the use of screencast (SN and STN) in the learning of C++ computer programming among local university students.

2. To investigate whether there is any significant difference in the students' achievement in learning of C++ computer programming between those using the screencast and narration (SN) mode, and screencast, text and narration (STN) mode.

3. To identify any significant difference in the students' attitude towards learning of C++ computer programming between those using the screencast and narration (SN) mode, and screencast, text and narration (STN) mode.

4. To investigate whether there is any significant interaction effects between the different levels of logic (Low/High) and the two learning modes which are screencast and narration (SN), and screencast, text and narration (STN) in the students' achievement in learning of C++ computer programming.

5. To identify the correlation between the students' attitude towards learning of C++ computer programming and the students' achievement in learning of C++ computer programming using the screencast and narration (SN) mode.

6. To identify the correlation between the students' attitude towards learning of C++ computer programming and the students' achievement in learning of C++ computer programming using the screencast, text and narration (STN) mode.

1.5 Research Questions

This study is design to answer the following questions:

1. Is there any significant difference in the students' achievement in learning C++ computer programming between those using the screencast and narration (SN) mode, and screencast, text and narration (STN) mode?

2. Is there any significant difference in the students' attitude in learning C++ computer programming between those using the screencast and narration (SN) mode, and screencast, text and narration (STN) mode?

3. Is there any significant interaction effects between the different levels of logic (Low/High) and the two learning modes, which are screencast and narration (SN), and screencast, text and narration (STN) in the students' achievement in learning C++ computer programming?

4. Is there any significant correlation between the students' attitude towards learning C++ computer programming and their achievement towards learning C++ computer programming using SN mode?

5. Is there any significant correlation between the students' attitude towards learning C++ computer programming and their achievement towards learning C++ computer programming using STN mode?

1.6 Research Hypotheses

This study aims to test the following null hypothesis whether to accept or reject the hypothesis based on the level of probability value set. The significance level of alpha is set at 0.05 and it is used to test the statistical significance.

$H_0$1: There is no significant difference in the students' achievement in learning C++ computer programming between those using the screencast and narration (SN) mode, and screencast, text and narration (STN) mode.

$H_0$2: There is no significant difference in the students' attitude in learning C++ computer programming between those using the screencast and narration (SN) mode, and screencast, text and narration (STN).

$H_0$3: There is no significant interaction effects between the different levels of logic (Low/High) and the two learning modes which are screencast and narration (SN), and screencast, text and narration (STN) in the students' achievement in learning C++ computer programming.

$H_0$4: There is no significant correlation between the students' attitude towards learning C++ computer programming and their achievement towards learning C++ computer programming in using the SN mode.

$H_0 5$: There is no significant correlation between the students' attitude towards learning C++ computer programming and their achievement towards learning C++ computer programming in using the STN mode.

1.7 Theoretical Framework

The underpinned theoretical framework of this research is based on Mayer's Cognitive Theory of Multimedia Learning (Mayer, 2005a; Mayer & Moreno, 2003). A brief explanation of the theory is given here and further elaboration is presented in Chapter 2.

According to Mayer's Cognitive Theory of Multimedia Learning, there are three assumptions, i.e. there are two separate channels for processing information, limited channel capacity, and that learning is an active process. The cognitive process involved in active learning requires the allocation of a certain amount of learning materials channelling to the sensory memory.

There are three important processes in active learning. It begins with the learner selecting relevant materials such as words and images. This is followed by the process of organising the words and images into verbal and visual models. Finally, the verbal and visual models are integrated with prior knowledge. Besides that, the human memory system is also divided into three main memory stores in multimedia learning and they are the sensory memory, working memory, and long-term memory (Mayer, 2005a).

Besides understanding the criteria to successfully implement multimedia learning environment, Sweller's cognitive load theory (Sweller, 1999) states that the focus on the limitations of a human's working memory at a specific learning duration also needs to be taken into consideration as well. According to the Cognitive Load Theory (CLT), there are three types of cognitive loads, mainly, intrinsic cognitive load, extraneous cognitive load, and germane cognitive load. Intrinsic cognitive load refers to the difficulty in the content of certain subjects or topics, which cannot be modified. Extraneous cognitive load refers to the instructional materials used to present information to the learner. Lastly, germane cognitive load refers to the increase in cognitive load due to cognitive activities in the working memory that exceed beyond simple tasks.

According to Cooper (1998), the effectiveness of learning depends on the intrinsic cognitive load, extraneous cognitive load, and mental resources. Learning will occur when the total cognitive load does not exceed the mental resources. This happens when the intrinsic cognitive load is low (simple content) and sufficient mental resources are available for the learners to learn from the instructional materials.

However, if the intrinsic cognitive load is high (difficult content) and the extraneous cognitive load is high as well, then the total cognitive load will exceed the mental resources. Thus, learning may fail to occur. Therefore, proper allocation of cognitive resources is critical in creating an effective learning environment and the design of instructional materials to reduce the extraneous cognitive load may enhance learning effectively.

Hence, the design and development of the two screencast (SN and STN) contents are based on Mayer's Cognitive Theory of Multimedia Learning (Mayer,

2005a; Mayer & Moreno, 2003) to avoid overloading of the limited capacity of the dual channels. Moreover, Sweller's cognitive load theory (Sweller, 1999) is adhere to reduce the total cognitive load by reducing the extraneous cognitive load to ensure that it does not exceed the mental resources.

1.8 Conceptual Framework

The research sample is made up of tertiary education students that have never taken any formal C++ programming courses before. They are randomly divided into two groups, which are the screencast and narration (SN) mode, and screencast, text, and narration (STN) mode. The topics learned by the sample is Introduction to C++ Computer Programming.

The independent variable consists of the two multimedia-based learning materials that are made up of the elements' screen recording, text, and narration. With the combination of these elements, two different types of multimedia learning modes were produced, namely (i) screencasting and narration (SN) mode and (ii) screencasting, text and narration (STN) mode. The SN mode is used to investigate the effects of the modality principle, whereas the STN mode is used to investigate the effects of the redundancy principle. Moreover, the research determines which multimedia learning modes is better in increasing the students' understanding of C++ computer programming based on their achievement. The two learning modes are also used to ascertain which mode is able to instil more positive attitude towards the learning of C++ computer programming based on the students' Post-Cas.

The moderator variable in this study is the different levels of logical thinking among the samples' populations. In order to divide the sample into low and high logical thinking groups, the sample was given the Group Assessment of Logical Thinking (GALT) test to determine their scores. Students that score equal or below 50% is classified as low logic group, whereas students that score above 50% is classified as high logic group.

The dependent variables are the students' achievements and attitude. The achievement of the sample is measured based on the C++ programming posttest score, while attitude is measured using the Computer Attitude Scale (CAS). To ascertain whether there is any relationship between one's attitude and their achievement, the Pearson correlation is conducted to measure the strength of the relationship between these two variables. Figure 1.1 shows the conceptual framework of this research.

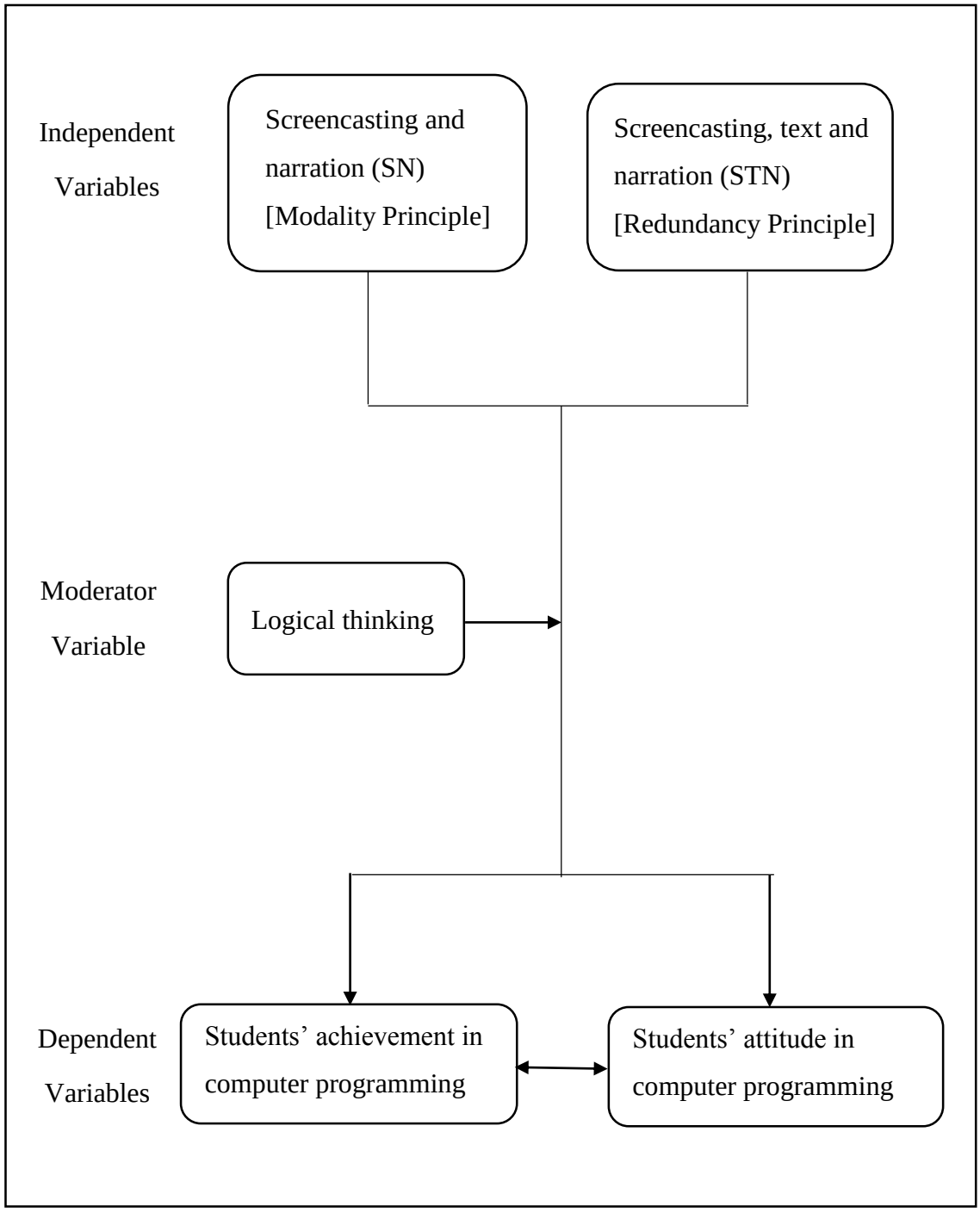

Figure 1.1 Conceptual Framework

1.9 Significance of the Study

The multimedia learning environment has been around for many years. However, the awareness on the effectiveness and usage has been low. Thus, this study hopes to create an awareness on the effectiveness and potential of interactive multimedia learning tools, with the use of screencast in the teaching and learning of C++ computer programming.

This study proposed to investigate the effectiveness of screencasting in the teaching of introductory programming on the achievement of students from different learning modes. Besides that, it is also to ascertain whether screencasting has created a positive or negative impact on the students' attitudes as well.

The study also aims at identifying what types of instructional activities are suitable and effective in learning computer programming. Besides that, it examines the impact of the redundancy and modality principles between the two learning modes. By knowing the reason behind these two principles, instructional designers will be able to develop more effective screencasting learning materials in the learning of C++ computer programming.

Furthermore, the researcher has adopted some multimedia principles and ASSURE design and development model. This research scrutinises the effectiveness of the design and development process as well as the quality of the screencast. The findings of this study will provide instructional designers with insight information regarding the type of design and development processes, teaching strategies, cognitive approaches, and motivation factors that will further guide multimedia developers to create a holistic and effective teaching material. Moreover, it will enable educators to

understand the psychological aspect of attitude in determining one's actions towards success or failure in performing a certain task.

By identifying the important aspects of developing an effective instructional material, it will accommodate the various learning styles of students in creating a conducive learning environment, thus increasing the understanding of C++ computer programming.

1.10 Limitations

Limitations are unavoidable in every research study. They are something that cannot be controlled by the researcher himself, and might be due to the environment, nature of the study, background of the samples, conditions or time constraints that might influence the findings of the study. Therefore, in this study a few limitations need to be highlighted.

The two modes of screencasting materials are based on limited topics in the introduction to C++ computer programming. Therefore, it is insufficient to conduct the effectiveness of the treatment on much more difficult topics. Besides that, the subject taught in this study is only based on the introduction to C++ computer programming. Thus, it will restrict the generalisation of the findings to other subjects' curriculum.

Furthermore, the research lasted for five weeks. At the beginning of week one, samples were briefed on the research and given the C++ programming pretest. Each week, they were given a new topic to learn and at the end of week five, the samples were then given the C++ programming posttest. Within the treatment period from